Journal of Ubiquitous Computing and Communication Technologies (ISSN: 2582-337X)
www.irojournals.com/jucct/

Detection of DDOS Attack using Decision Tree Classifier in SDN Environment

Nithish Babu S[1], Yogesh V[2], Mariswaran S[3], Gowtham N[4]

[1,2,3]B.Tech Information Technology, National Engineering College Kovilpatti, Tamilnadu, India

[4]Assistant Professor Information Technology, National Engineering College Kovilpatti, Tamilnadu, India

E-mail: [1]1915011@nec.edu.in, [2]1915019@nec.edu.in, [3]1915022@nec.edu.in, [4]gowthamiit@nec.edu.in

Abstract

Software Defined Networking (SDN) is a dynamic architecture that employs a variety of applications for making networks more adaptable and centrally controlled. It is easy to attack the entire network in SDN because the control plane and data plane are separated. DDoS attack is major danger to SDN service providers because it can shut down the entire network and stop services to all customers at any time. One of the key flaws of most SDN architectures is lack of susceptibility to DDoS attacks with its types like TCP flooding, UDP flooding, SYN flooding, ICMP flooding and DHCP flooding for detecting those kinds of attacks. The machine learning algorithms are widely used in recent years to identify DDoS attacks. This research utilizes Decision Tree Classifier for detection and classification of DDoS attacks on SDN. The Forward Feature Selection technique is also used in the research to select the best features from the dataset and from that dataset the data are employed to train and test the model by Decision Tree Classifier Algorithm. The decision Tree Classifier technique is a supervised method used to forecast desired values of observations using rudimentary machine learning decision rules derived from training data. Based on the accuracy of decision tree techniques, in future, a hybrid learning model will be designed for detecting the Distributed Denial of Services in an SDN environment with high accuracy and a low false negative rate.

Keywords: SDN, DDOS, Decision Tree Classifier, Feature selection, Forward feature selection.

DOI: https://doi.org/10.36548/jucct.2023.2.006
Received: 03.05.2023, received in revised form: 03.06.2023, accepted: 18.06.2023, published: 30.06.2023

1. Introduction

In order to improve the performance of the network and monitoring, software defined networking (SDN) is a network management method that offers dynamic and programmatically effective network architecture. Compared to traditional network management, cloud computing. SDN aims to deal with static architecture. By separating the routing mechanism (control plane) from network packet forwarding (data plane), it tries to consolidate functionality into a single network component. One or more controllers make up the control plane and serve as the SDN network's brain because they are its intellect. The issue with SDN is that it is centralized, and has drawbacks in terms of security, scalability, and elasticity. The OpenFlow protocol is a Software Defined Network architecture. A DDoS uses a botnet, or network of connected online devices, to saturate a target website with traffic. The whole internet user base can be blocked by a successful distributed denial of service attacks, which is a very notable occurrence.

2. Objective

The major goal is to identify DDOS in the network. The best features will be selected from the dataset using forward feature selection algorithm. The trained system is used to detect the nature of incoming TCP traffic pattern, ie. Normal request/abnormal request in SDN with high accuracy.

3. Methodology

Feature Selection is choosing the most pertinent characteristics such as variables or predictors from a selection redefined data set and utilized to construct the model. Engineers in this field of machine learning and data mining find data analysis tough when there are high dimension (N) number of features. By eliminating of redundant and irrelevant data, feature selection offers a practical solution to this type of issue. This can speed up computation, increase learning accuracy, and provide a deeper comprehension of the learning model or data. For training the model, dataset is taken from “Kaggle (Link: https://www.kaggle.com/datasets/aikenkazin/ddos-sdn-dataset)”. This works by including or

excluding important features without changing them. Fig 3.1 shows various feature selection techniques in the machine learning.

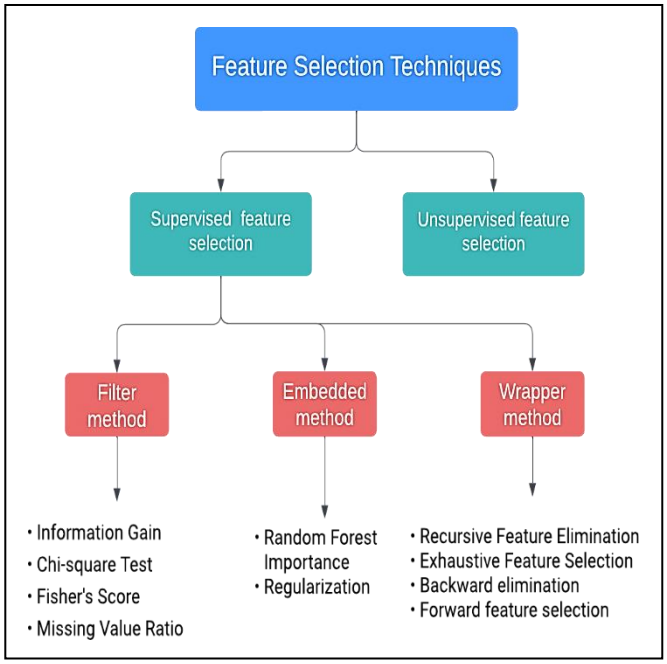

Figure 3.1. Various Feature Selection Techniques

A. Filter Method

Instead of cross-validation performance, filter methods picks univariate statistics are used to quantify intrinsic qualities of features. These methods are less costly and quicker than wrapper approaches.

B. Wrapper Method

Wrappers need a way to search through all potential feature subset subgroups and evaluate the quality of each feature subset by learning and also evaluating classifier using that feature subset. A certain machine learning algorithm attempting to fit on given dataset forms the basis of the feature selection procedure. By comparing every conceivable feature combination to the evaluation criterion, it employs a greedy search methodology. In most cases, wrapper approaches outperform filter methods in terms of prediction accuracy.

C. Embedded Method

These techniques incorporate feature interactions while preserving minimal computational costs, combining the benefits of wrapper and filter methods. Embedded techniques are iterative in that they handle each iteration procedure of model training also carefully extract the features that contribute the maximum to train for that iteration.

D. Differentiate of Three Methods

Filter method works faster when there are many features. Wrapper method gives better performance and embedded method lies in between the other two methods. Wrapper approaches evaluate the value of features based on the performance of the classifier. Filter methods uses relevance properties of features measured by univariate statistics. Wrapper approaches are therefore improving classifier performance, but because they require more cross-validation and additional learning steps than filter methods, they are also more expensive. Because embedded techniques are also employed to enhance the functionality of learning algorithms or models, they resemble wrapper methods in many ways. In contrast to wrapper techniques, learning involves the application of an intrinsic model-building metric. By these differentiation study forward feature selection method is chosen in the wrapper method.

4. Forward Feature Selection

Huseyin Polat, Onur Polat and Aydin Cetin [9] By referring to the paper the Wrapper method has more accuracy, Sensitivity, and Specificity Precision than the Filter method and Embedded method. The forward feature selection algorithm works by starting with an empty set of features and adding features one at a time to this set until all features have been added. At each step, the feature that results in the largest improvement to the performance of the model is added to the set. The primary benefit of this method that it is guaranteed to find the set of features that results in the best performance of the model and it can be computationally expensive, especially if there are a large number of features. Initially, Forward Stepwise selection uses a null model. i.e. begins with the model's first variable. The top model among the best models in each k is then selected based on RSS, CV, or adjusted R square. Next, predictors (Features) are added one at a time. Once that predictor is chosen in this procedure,

it never deviates in the second stage. This is carried out repeatedly until the ideal subset of the 'k' predictor features is chosen. Forward selection is restricted by fact that a model-included predictor never declines. Therefore, in forward selection, the selection models are 1+N(N+1)/2. When N=10, the total models in subset selection were 1024, but in forward selection, the computational capacity is reduced, therefore the total models in this approach are only 211. subsets of data are created to train a model. The features are added and removed and the model is retrained based on the model's output. It employs a greedy strategy to create the subgroups and assesses the accuracy of all potential feature combinations.

5. Dataset

The TCP dataset is taken from Kaggle which is the SDN context, utilized to detect DDOS attacks. The dataset contains 29,436 records and 20 features and from that 80% of dataset which is 23,548 records used to train the model by selecting best features in the dataset using forward feature selection technique and 20% of dataset which is of 5,888 used to test the Decision Tree Classifier trained model.

6. Decision Tree

The decision Tree algorithm is a supervised learning algorithm family. In contrast to other methods, decision tree technique can handle both classification and regression problems. A strong and well-liked categorization and prediction tool is a decision tree. The Decision nodes and leaf nodes, which stand for categories that tree may categories, make up the Decision Tree. A decision tree can also be used to depict a flowchart, in which the decision is taken at the leaves at the end of the flow from the root node. When anticipating the class label of a record, we begin at tree's root. The values of the root attribute to those the attribute on the record are compared. Based on comparison, the branch corresponding to the value is followed and jumped to next node. Fig 6.1 shows the decision tree

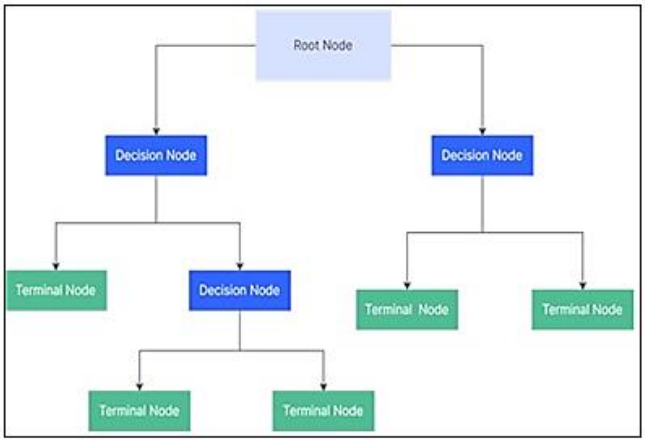

Figure 6.1. Flow Diagram of Decision Tree Classifier

The decision criteria for regression and classification trees are different. To determine whether to divide a node into two or more sub-nodes, it uses a variety of techniques. The homogeneity of freshly produced sub-nodes is improved through sub node development. It employs a number of methods to decide whether to split a node into two or more sub-nodes. Through the development of sub nodes, the homogeneity of recently created sub-nodes is enhanced.

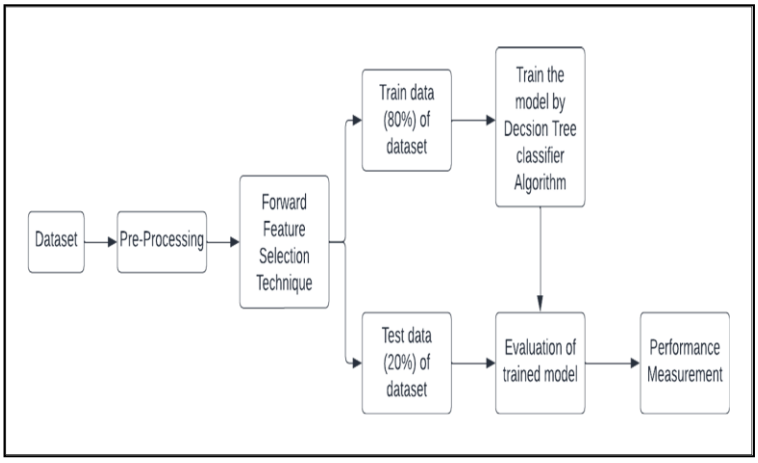

Figure 6.2. Shows the Flow Diagram of Detection of DDOS Attack

Fig 6.2 represents the flow diagram detection of DDOS attacks in SDN environments. The dataset taken from Kaggle is pre-processed so that null values and floating values will be removed. By using Forward feature selection technique, best features were selected from the dataset. The model is trained using 80% data from the best features and decision tree algorithm. The 20% of data is used as train data to evaluate the trained model. The

performance measure of Accuracy, Recall, Precision, F1_score is concluded as the experimental result.

7. Result & Findings

In this experiment, the dataset was used to examine data from DDoS attack traffic during normal and abnormal traffic periods on SDN architecture. To train and evaluate the model, feature selection approaches were utilized.

$$Accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)},$$
$$Precision = \frac{(TP)}{(TP + FP)},$$
$$F1 - Score = \frac{(2 * TP)}{(2 * TP + FP + FN)},$$
$$Recall = \frac{(TP)}{(TP + FN)},$$

The topology contains a RYU Controller(c0) which implements the behavior of a learning switch, OpenFlow Switches (s1,s2,s3,s4,s5) expected to be connected to a controller and "Hosts(h1,h2,h3,h4,h5,h6,h7,h8,h9,h10)". The hosts (h1&h2) were connected to the switch (s1). (h3,h4,h5) were connected to the switch (s2). (h6) connected to switch(s3). (h7) connected to switch(s4). (h8,h9,h10) were connected to switch(s5). Fig 7.1 shows the linear topology created using the mininet and miniedit.

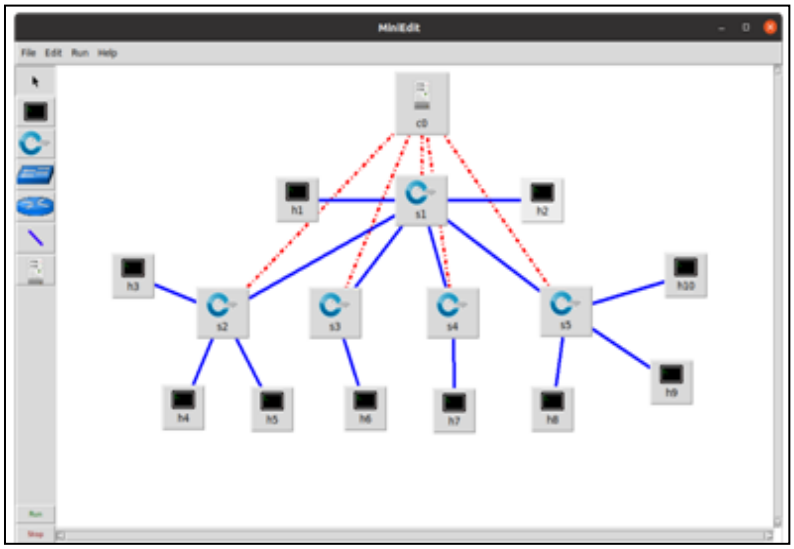

Figure 7.1. Linear Topology

8. Conclusion And Future Work

Although SDN has many benefits, it is susceptible to DDoS attacks, which is the main network security issue. Centralized control is a feature of SDN, but it also increases SDN controller's vulnerability in response to security risks like DDoS attacks. Features selected from the dataset by using forward feature selection algorithm of machine learning. In this study a linear network is topology is created using the mininet and miniedit.

In future, the Decision Tree Algorithm will be embedded in the SDN controller and that will classify the request from user is normal user or abnormal user. The performance of Decision tree algorithm will the compared with one more supervised machine learning algorithm which provides high accuracy in the detection rate of the DDOS attack.

References

[1] Yungaicela-Naula, Noe M., Cesar Vargas-Rosales, Jesus Arturo Perez-Diaz, and Diego Fernando Carrera. "A flexible SDN-based framework for slow-rate DDoS attack mitigation by using deep reinforcement learning." Journal of Network and Computer Applications (2022): 103444.

[2] Wang, Song, Juan Fernando Balarezo, Karina Gomez Chavez, Akram Al-Hourani, Sithamparanathan Kandeepan, Muhammad Rizwan Asghar, and Giovanni Russello. "Detecting flooding DDoS attacks in software defined networks using supervised learning techniques." Engineering Science and Technology, an International Journal (2022): 101176.

[3] Sangodoyin, Abimbola O., Mobayode O. Akinsolu, Prashant Pillai, and Vic Grout. "Detection and Classification of DDoS Flooding Attacks on Software-Defined Networks: A Case Study for the Application of Machine Learning." IEEE Access 9 (2021): 122495- 122508.

[4] Aljuhani, Ahamed. "Machine learning approaches for combating distributed denial of service attacks in modern networking environments." IEEE Access 9 (2021): 42236- 42264.

[5] Sudar, K. Muthamil, M. Beulah, P. Deepalakshmi, P. Nagaraj, and P. Chinnasamy. "Detection of Distributed Denial of Service Attacks in SDN using Machine learning techniques." In 2021 International Conference on Computer Communication and Informatics (ICCCI), pp. 1-5. IEEE, 2021.

[6] Yungaicela-Naula, Noe Marcelo, Cesar Vargas-Rosales, and Jesus Arturo Perez-Diaz. "SDN-based architecture for transport and application layer DDoS attack detection by using machine and deep learning." IEEE Access 9 (2021): 108495-108512.

[7] Ahuja, Nisha, Gaurav Singal, Debajyoti Mukhopadhyay, and Neeraj Kumar. "Automated DDOS attack detection in software defined networking." Journal of Network and Computer Applications 187 (2021): 103108.

[8] Scaranti, Gustavo F., Luiz F. Carvalho, Sylvio Barbon, and Mario Lemes Proenca. "Artificial immune systems and fuzzy logic to detect flooding attacks in software-defined networks." IEEE Access 8 (2020): 100172-100184.

[9] Polat, Huseyin, Onur Polat, and Aydin Cetin. "Detecting DDoS attacks in software-defined networks through feature selection methods and machine learning models." Sustainability 12, no. 3 (2020): 1035.

[10] Chen, Yixin, Jianing Pei, and Defang Li. "DETPro: a high-efficiency and low-latency system against DDoS attacks in SDN based on decision tree." In ICC 2019-2019 IEEE International Conference on Communications (ICC), pp. 1-6. IEEE, 2019.

[11] Glavan, Dragos, Ciprian Racuciu, Radu Moinescu, and Narcis-Florentin Antonie. "Detecting the DDoS attack for SDN Controller." Scientific Bulletin" Mircea cel Batran" Naval Academy 22, no. 1 (2019): 1-8.

[12] Preamthaisong, Parinya, Anucha Auyporntrakool, Phet Aimtongkham, Titaya Sriwuttisap, and Chakchai So-In. "Enhanced DDoS detection using hybrid genetic algorithm and decision tree for SDN." In 2019 16th International Joint Conference on Computer Science and Software Engineering (JCSSE), pp. 152-157. IEEE, 2019.

[13] Wang, Meng, Yiqin Lu, and Jiancheng Qin. "A dynamic MLP-based DDoS attack detection method using feature selection and feedback." Computers & Security 88 (2020): 101645.

[14] El Sayed, Mahmoud Said, Nhien-An Le-Khac, Marianne A. Azer, and Anca D. Jurcut. "A Flow-Based Anomaly Detection Approach With Feature Selection Method Against DDoS Attacks in SDNs." IEEE Transactions on Cognitive Communications and Networking 8, no. 4 (2022): 1862-1880.

[15] Chartuni, Andrés, and José Márquez. "Multi-classifier of DDoS attacks in computer networks built on neural networks." Applied Sciences 11, no. 22 (2021): 10609.

[16] Tonkal, Özgür, Hüseyin Polat, Erdal Başaran, Zafer Cömert, and Ramazan Kocaoğlu. "Machine learning approach equipped with neighbourhood component analysis for ddos attack detection in software-defined networking." Electronics 10, no. 11 (2021): 1227.

[17] Alashhab, Abdussalam Ahmed, Mohd Soperi Mohd Zahid, Mohamed A. Azim, Muhammad Yunis Daha, Babangida Isyaku, and Shimhaz Ali. "A Survey of Low Rate DDoS Detection Techniques Based on Machine Learning in Software-Defined Networks." Symmetry 14, no. 8 (2022): 1563.